IN THE SPECIFICATION

Please amend the title to read:

Providing An Adder With A Conversion Circuit In A Slack Propagation Path

~~Providing A Fast Adder For Processor Based Devices~~

Please amend the paragraph beginning on page 3, line 12, as follows:

The general concept of carry arbitration will be described to aid in the understanding of the invention. In the general case, the carry c_{i+1} is evaluated by adding two 1-bit binary numbers a_i and b_i. There are two general cases defined by the values of a_i and b_i. The first case, where there is an output carry request, arises when both operand bits are equal. A 1-carry request occurs if both inputs are 1, whereas a 0-carry request if both inputs are 0. The second case, where there is no output carry request, arises when the operand bits have different values. See Table 1 in which the letter u indicates there is no output carry request.

Please amend Table 2 beginning on page 5, line 1, as follows:

TABLE 2

a_ib_i	a_i,b_i ~~a_i,b_i~~	C_{i+1}
0 0	- -	0
1 1	- -	1
0 1 (or 1 0)	0 0	0
0 1 (or 1 0)	1 1	1
0 1 (or 1 0) ~~(opr 1 0)~~	0 1 (or 1 0)	u

Please amend Table 3 beginning on page 5, line 5, as follows:

Table 3

C_{i+1}	V_i,W_i ~~V_i,W_i~~
0	0 0
1	1 1
u	0 1
u	1 0

Please amend the paragraph beginning on page 6, line 15, as follows:

The following equations satisfy Tables 3 and 4:

$V_i = a_ib_i + (a_i + b_i)(a_jb_j + (a_j + b_j)a_k)$ (3)

$\underline{W_i}$ ~~V_i~~ $= a_ib_i + (a_j + b_i)(a_jb_j + (a_j + b_j)b_k)$

Please amend the paragraph beginning on page 13, line 24, as follows:

A block diagram of an improved 80-bit adder is shown in Figure 12. The whole adder is visualized (but not divided) as consisting of five 16-bit groups. The first row is the conversion circuit, which contains 2-input AND ~~NAND~~ and OR ~~NOR~~ gates 1201-1209. The second and third rows are four-way arbiters that produce carries within each group and have the form discussed previously. The fourth row, 1211-1219, produces two intermediate sums with a zero carry-in and a one carry-in. The final row includes multiplexers which select the final sum result and three carry arbiters which generate the boundary carries c_{16}, c_{32}, c_{48} and c_{64}. The carries of the 16 least significant bits have already been generated after two rows of the carry computation. Compared with the conventional carry-select scheme, the need for group adders has been eliminated. The two intermediate sums are elegantly generated within the carry generation tree. This may result in a significant reduction of chip area, especially when the groups are made to be long, since group adders also need some mechanisms for carry computation.

Please amend the paragraph beginning on page 18, line 8, as follows:

As described in association with Figure 14, by moving the NAND ~~AND~~ and NOR ~~OR~~ gates 1201 – 1209 into the slack path of the 4th row (with respect to the adder of Figure 12), the total propagation time through the adder may be reduced. The critical timing path may be reduced from four rows to three rows.